UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/657,121	09/07/2000	Randal Craig Swanberg	AUS9-1999-0305-US1	2398

7590 03/15/2004

Kelly K Kordzik
Suite 800
100 Congress Avenue
Austin, TX 78701

EXAMINER	
PARTHASARATHY, PRAMILA	
ART UNIT	PAPER NUMBER
2136	2

DATE MAILED: 03/15/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	**Application No.** 09/657,121	**Applicant(s)** SWANBERG ET AL.
	Examiner Pramila Parthasarathy	**Art Unit** 2136

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *07 September 2000*.

2a)□ This action is **FINAL**. 2b)☒ This action is non-final.

3)□ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-36* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)□ Claim(s) _____ is/are allowed.

6)□ Claim(s) *1-36* is/are rejected.

7)□ Claim(s) _____ is/are objected to.

8)□ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)□ The specification is objected to by the Examiner.

10)□ The drawing(s) filed on _____ is/are: a)□ accepted or b)□ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)□ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)□ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)□ All b)□ Some * c)□ None of:

1.□ Certified copies of the priority documents have been received.

2.□ Certified copies of the priority documents have been received in Application No. _____.

3.□ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)

2) □ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) □ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
Paper No(s)/Mail Date _____.

4) □ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ .

5) □ Notice of Informal Patent Application (PTO-152)

6) □ Other: _____.

DETAILED ACTION

1. This action is in response to the application filed on 09/07/2000. Claims 1 – 36 were received for consideration. No preliminary amendments to the claims were filed. Claims 1 – 36 are currently being considered.

Claim Objections

2. Claim 1 is objected to because of the following informalities:

Missing period at the end of the claim 1.

Claim 24 is objected to because of the following informalities:

Missing period at the end of the claim 24.

Appropriate corrections are required.

Claim Rejections - 35 USC § 102

The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

3. Claims 1 – 5, 7, 9 – 13, 15 – 20, 22 – 28 and 32 – 36 are rejected under 35 U.S.C. 102(b) as being anticipated by Kane et al. (U.S. Patent No: 5,596,739).

Regarding Claim 1, Kane teaches and describes a method for stack memory protection (Fig. 1) comprising the steps of:

generating new memory page attributes for a page table used to manage memory, each of said new memory page attributes identifying a block memory as a new class of memory, each of said new memory page attributes generated by a corresponding new load/store instruction (Fig.1, Column 1 lines 40 – 46 and 56 – 59);

assigning, by an operating system or a processor, a selected one of said new memory page attributes to a selected block of memory, said selected block of memory used as a new class of memory corresponding to said selected new memory page attribute (Column 1 lines 56 – 59);

blocking normal load/stores to a memory block having one of said new memory page attributes (Column 2 lines 8 – 11); and

blocking a first new load/store to a memory block with one of said new memory page attributes not corresponding to said first new load/store (Column 2 lines 2 – 11).

Regarding Claim 10, Kane teaches and describes a processor comprising stack memory protection circuitry, said processor using blocks of memory as stack memory, said stack memory protection circuitry comprising:

a stack memory attribute circuit, said stack memory attribute circuit operable to generate memory attributes, said memory attributes associated with each memory block designated as a memory stack (Fig.1 Column 1 lines 40 – 46 and 56 – 59);

a page table attribute storage circuit, said page table attribute circuit operable to store and associate one of said stack memory attributes, said stack memory attributes with a block of memory designated as stack memory (Fig. 2A #330 and Column 5 lines 6 – 16);

a stack memory allocation circuit, said stack memory allocation circuit operable to identify a block of memory as a stack memory and associate said memory block with one of said stack memory attributes, said stack memory attributes stored in a memory page table (Column 1 lines 56 – 59); and

a stack memory instruction execution circuit, said stack memory instruction execution circuit operable to decode load/store instructions to memory blocks, said stack memory instruction execution circuit granting stack memory load and stores to memory blocks having a required stack memory attribute and not granting stack memory load and stores to memory blocks not having said required stack memory attribute (Column 3 lines 18 – 23, Column 14 lines 36 – 42 and Column 2 lines 2 – 11).

Regarding Claim 17, Kane teaches and describes a data processing system, comprising a central processing unit (CPU) (Column 3 line 4 – 39);

shared random access memory (RAM) (Column 3 line 4 – 39);

read only memory (ROM) (Fig. 2 Column 3 lines 4 – 39);

an I/O adapter (Column 3 lines 4 – 39); and

a bus system coupling said CPU to said ROM, said RAM said display adapter, wherein said CPU, said CPU comprising stack memory protection circuitry (Column 4 lines 57 – 62), said stack memory protection circuitry (Fig.1) comprising:

a stack memory attribute circuit, said stack memory attribute circuit operable to generate memory attributes, said memory attributes associated with each memory block designated as a memory stack (Fig.1, Column 1 lines 40 – 46 and 56 – 59);

a page table attribute storage circuit, said page table attribute circuit operable to store and associate one of said stack memory attributes, said stack memory attributes with a block of memory designated as stack memory (Fig.2A # 330 and Column 5 lines 6 – 16);

a stack memory allocation circuit, said stack memory allocation circuit operable to identify a block of memory as a stack memory and associate said memory block with a stack memory attributes, said stack memory attributes stored in a memory page table (Column 1 lines 56 – 59); and

a stack memory instruction execution circuit, said stack memory instruction execution circuit operable to decode load/store instructions to memory blocks, said stack memory instruction execution circuit granting stack memory load and stores to memory blocks having a stack memory attribute and not granting stack memory load and stores to memory blocks not having said stack memory attribute (Column 3 lines 18 – 23, Column 14 lines 36 – 42 and Column 2 lines 2 – 11).

Regarding Claim 24, Kane teaches and describes a computer program product embodied in a machine readable medium, including an operating system and a complier for a processor system (Fig.1), comprising: a program of instructions for performing the program steps of:

generating new memory page attributes for a page table used to manage memory, each of said new memory page attributes identifying a block memory as a new class of memory, each of said new memory page attributes generated by a corresponding new load/store instruction (Fig.1 and Column 1 lines 40 – 46 and 56 – 59);

assigning, by an operating system or a processor, a selected one of said new memory page attributes to a selected block of memory, said selected block of memory used as a new class of memory corresponding to said selected new memory page attribute (Column 1 56 – 59);

blocking normal load/stores to a memory block having one of said new memory page attributes (Column 2 lines 8 – 11); and

blocking a first new load/store to a memory block with one of said new memory page attributes not corresponding to said first new load/store (Column 2 lines 2 – 11).

Regarding Claim 33, Kane teaches and describes a method of managing a memory device comprising the steps of:

partitioning said memory device into a plurality of memory spaces on an as-needed bases (Column 1 lines 56 – 59); and associating memory attribute with each memory space; said memory attribute determining a use of each of said memory spaces (Column 1 lines 56 – 59).

Claim 2 is rejected as applied above in rejecting claim 1. Furthermore, Kane teaches and describes a method for stack memory protection, wherein: said new classes of memory comprises stack memory (Column 8 lines 8 – 17).

Claim 11 is rejected as applied above in rejecting claim 10. Furthermore, Kane teaches and describes a processor comprising stack memory protection circuitry, wherein: a first error condition is generated whenever normal load/stores are attempted to stack memory having a first or a second stack memory attribute (Column 13 lines 66 – Column 14 line 3).

Claim 12 is rejected as applied above in rejecting claim 10. Furthermore, Kane teaches and describes a processor comprising stack memory protection circuitry, wherein: a second error condition is generated whenever said stack memory load/stores are attempted to memory not having a stack memory attribute (Column 13 lines 66 – Column 14 line 3).

Claim 13 is rejected as applied above in rejecting claim 10. Furthermore, Kane teaches and describes a processor comprising stack memory protection circuitry, wherein: a third error condition is generated whenever a stack memory load/store for a first memory stack is attempted to a second memory stack, said third error condition also generated if a stack memory load/store for said second memory stack is attempted to said first memory stack (Column 13 lines 66 – Column 14 line 3).

Claim 18 is rejected as applied above in rejecting claim 17. Furthermore, Kane teaches and describes a data processing system, wherein: a first error condition is generated whenever normal load/stores are attempted to stack memory having a first or a second stack memory attribute (Column 13 lines 66 – Column 14 line 3).

Claim 19 is rejected as applied above in rejecting claim 17. Furthermore, Kane teaches and describes a data processing system, wherein: a second error condition is generated whenever said stack memory load/stores are attempted to memory not having a stack memory attribute (Column 13 lines 66 – Column 14 line 3).

Claim 20 is rejected as applied above in rejecting claim 17. Furthermore, Kane teaches and describes a data processing system, wherein: a third error condition is generated whenever a stack memory load/store for a first memory stack is attempted to a second memory stack, said third error condition also generated if a stack memory load/store for said second memory stack is attempted to said first memory stack (Column 13 lines 66 – Column 14 line 3).

Claim 25 is rejected as applied above in rejecting claim 24. Furthermore, Kane teaches and describes a computer program product embodied in a machine readable medium, wherein: said new classes of memory comprise stack memory (Column 8 lines 8 – 17).

Claim 34 is rejected as applied above in rejecting claim 33. Furthermore, Kane teaches and describes a method of managing a memory device, wherein: a particular memory attribute has corresponding load/store instruction (Fig. 1 #114 and Column 2 lines 33 – 37).

Claim 36 is rejected as applied above in rejecting claim 33. Furthermore, Kane teaches and describes a method of managing a memory device, wherein: each of said memory attributes are stored in a memory page table, said memory page table used to manage said memory device (Column 1 lines 64 – 66).

Claim 3 is rejected as applied above in rejecting claim 2. Furthermore, Kane teaches and describes a method for stack memory protection, wherein: a first error condition is generated whenever normal load/stores are attempted to stack memory having a first or a second stack memory attribute (Column13 lines 66 – Column 14 line 3).

Claim 4 is rejected as applied above in rejecting claim 2. Furthermore, Kane teaches and describes a method for stack memory protection, wherein: a second error condition is generated whenever said stack memory load/stores are attempted to memory not having a stack memory attribute (Column 13 lines 66 – Column 14 line 3).

Claim 5 is rejected as applied above in rejecting claim 2. Furthermore, Kane teaches and describes a method for stack memory protection, wherein: a third error condition is generated whenever a stack memory load/store for a first memory stack is attempted to a second memory stack, said third error condition also generated if a stack memory load/store for said second memory stack is attempted to said first memory stack (Column 13 lines 66 – Column 14 line 3).

Claim 15 is rejected as applied above in rejecting claim 13. Furthermore, Kane teaches and describes a processor comprising stack memory protection circuitry, wherein: said first memory stack is a processor stack, said processor stack used by a processor to load and store hardware register contents during program execution, said processor stacks transparent to a programmer or a compiler (Column 8 lines 3 – 32).

Claim 16 is rejected as applied above in rejecting claim 13. Furthermore, Kane teaches and describes a processor comprising stack memory protection circuitry, wherein: said second memory stack is a program stack, said program stack used by a programmer or a compiler in managing program flow (Column 8 lines 25 – 32).

Claim 22 is rejected as applied above in rejecting claim 20. Kane teaches and describes a data processing system, wherein: said first memory stack is a processor stack, said processor stack used by a processor to load and store hardware register contents during program execution, said processor stacks transparent to a programmer or a compiler (Column 8 lines 3 – 32).

Claim 23 is rejected as applied above in rejecting claim 20. Furthermore, Kane teaches and describes a data processing system, wherein: said second memory stack is a program stack, said program stack used by a programmer or a compiler in managing program flow (Column 8 lines 25 – 32).

Claim 26 is rejected as applied above in rejecting claim 25. Furthermore, Kane teaches and describes a computer program product embodied in a machine readable medium, wherein: a first error condition is generated whenever normal load/stores are attempted to stack memory having a first or a second stack memory attribute (Column 13 lines 66 – Column 14 line 3).

Claim 27 is rejected as applied above in rejecting claim 25. Furthermore, Kane teaches and describes a computer program product embodied in a machine readable medium, wherein: a second error condition is generated whenever said stack memory load/stores are attempted to memory not having a stack memory attribute (Column 13 lines 66 – Column 14 line 3).

Claim 28 is rejected as applied above in rejecting claim 25. Furthermore, Kane teaches and describes a computer program product embodied in a machine readable medium, wherein: a third error condition is generated whenever a stack memory load/store for a first memory stack is attempted to a second memory stack, said third error condition also generated if a stack memory load/store for said second memory stack is attempted to said first memory stack (Column 13 lines 66 – Column 14 line 3).

Claim 35 is rejected as applied above in rejecting claim 34. Furthermore, Kane teaches and describes a method of managing a memory device, wherein: a load/store instruction associated with a first memory attribute causes an error condition if attempted on a memory space with a second memory attribute (Column 1 Lines 64 – Column 2 line 8).

Claim 7 is rejected as applied above in rejecting claim 5. Furthermore, Kane teaches and describes a method for stack memory protection, wherein: said first memory stack is a processor stack, said processor stack used by a processor to load and store hardware register contents during program execution, said processor stacks transparent to a programmer or a compiler (Column 8 lines 3 – 17).

Claim 9 is rejected as applied above in rejecting claim 5. Furthermore, Kane teaches and describes a method for stack memory protection, wherein: said second memory stack is a program stack, said program stack used by a programmer or a compiler in managing program flow (Column 8 lines 25 –32).

Claim 32 is rejected as applied above in rejecting claim 28. Furthermore, Kane teaches and describes a computer program product embodied in a machine readable medium, wherein: said second memory stack is a program stack, said program stack used by a programmer or a compiler in managing program flow (Column 8 lines 3 – 17).

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

4. Claims 6, 8, 14, 21 and 29 – 31 are rejected under 35 U.S.C. 103(a) as being unpatentable over Kane et al (U S Patent 5,596,739) in view of Gillespie et al. (U S. Patent No. 5,657,475).

Claim 14 is rejected as applied above in rejecting claim 10. Furthermore, Kane teaches a processor comprising stack memory protection circuitry but does not explicitly teach said stack memory load and store instructions are executed on a CPU comprises an IA64 architecture. However Gillespie discloses stack memory load and store instructions are executed in parallel (IA-64 architecture) to the multi-ported registers (Column 3 lines 50 – 63). Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to implement a method for implementing a stock memory protection circuitry with stack memory load and store instructions to be executed on an IA64 architecture to increase the number of instructions that can be executed in parallel, which boosts performance. The motivation would have been to provide IA-64's massive resources; inherent scalability, explicit parallelism and full compatibility that allows high-performance.

Claim 21 is rejected as applied above in rejecting claim 17. Furthermore, Kane teaches and describes a data processing system, but does not explicitly teach stack memory load and store instructions are executed on a CPU comprising an IA64 architecture. However Gillespie discloses stack memory load and store instructions are executed in parallel (IA-64 architecture) to the multi-ported registers (Column 3 lines 50

– 63). Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to implement a method for implementing a stock memory protection circuitry with stack memory load and store instructions to be executed on an IA64 architecture to increase the number of instructions that can be executed in parallel, which boosts performance. The motivation would have been to provide IA-64's massive resources; inherent scalability, explicit parallelism and full compatibility that allows high-performance.

Claim 6 is rejected as applied above in rejecting claim 2. Furthermore, Kane teaches and describes a method for stack memory protection, but does not explicitly teach said stack memory load and store instructions are executed on a CPU comprises an IA64 architecture. However Gillespie discloses stack memory load and store instructions are executed in parallel (IA-64 architecture) to the multi-ported registers (Column 3 lines 50 – 63). Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to implement a method for implementing a stock memory protection circuitry with stack memory load and store instructions to be executed on an IA64 architecture to increase the number of instructions that can be executed in parallel, which boosts performance. The motivation would have been to provide IA-64's massive resources; inherent scalability, explicit parallelism and full compatibility that allows high-performance.

Claim 29 is rejected as applied above in rejecting claim 25. Furthermore, Kane teaches a computer program product embodied in a machine readable medium but does not explicitly teach said stack memory load and store instructions are executed on a CPU comprises an IA64 architecture. However Gillespie discloses stack memory load and store instructions are executed in parallel (IA-64 architecture) to the multi-ported registers (Column 3 lines 50 – 63). Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to implement a method for implementing a stock memory protection circuitry with stack memory load and store instructions to be executed on an IA64 architecture to increase the number of instructions that can be executed in parallel, which boosts performance. The motivation would have been to provide IA-64's massive resources; inherent scalability, explicit parallelism and full compatibility that allows high-performance.

Claim 30 is rejected as applied above in rejecting claim 29. Furthermore, Kane teaches and describes a computer program product embodied in a machine readable medium, wherein: said first memory stack is a processor stack, said processor stack used by a processor to load and store hardware register contents during program execution, said processor stacks transparent to a programmer or a compiler (Column 8 lines 3 – 17).

Claim 8 is rejected as applied above in rejecting claim 7. Gillespie discloses, a method for stack memory protection wherein: said processor stack is an IA64 register stack (Column 3 lines 50 – 63).

Claim 31 is rejected as applied above in rejecting claim 30. Furthermore, Gillespie discloses a method for managing a memory device, wherein: said processor stack is an IA64 register stack (Column 3 lines 50 – 63).

Conclusion

6. Any response to this action should be mailed to:

Commissioner of Patents and Trademarks, Washington, D.C. 20231 **or faxed to:** (703) 872-9306 for all formal communications.

Hand-delivered responses should be brought to Crystal Park II, 2121 Crystal Drive, Arlington, VA, Fourth Floor (Receptionist).

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Pramila Parthasarathy whose telephone number is 703-305-8912. The examiner can normally be reached on 8:00a.m. To 5:00p.m..

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Ayaz Sheikh can be reached on 703-305-9648. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to the receptionist whose telephone number is 703-305-3900.

Pramila Parthasarathy
Patent Examiner
703-305-8912
February 27, 2004

AYAZ SHEIKH
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100